A Comparison Between J2EE/EJB and Microsoft .NET

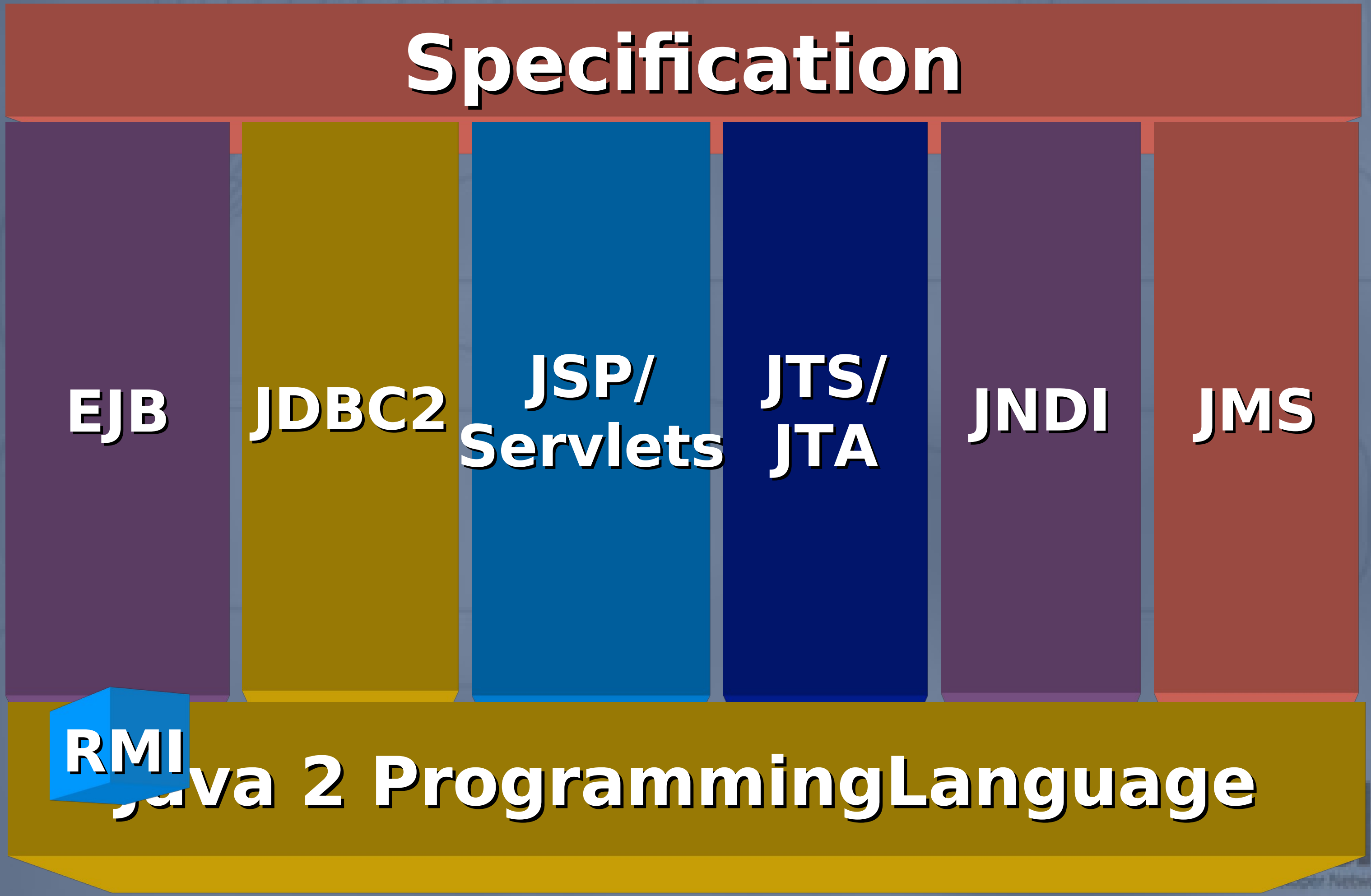
Carlos McKinley
Software Developer
NSP Technology Team
CarlosMc@Microsoft.com
Microsoft Corporation

Agenda

- **Architectures**
- **Components And Services**
- **Distributed Applications**
- **Accessing Data**
- **Web Services**
- **Web Applications**
- **Application Integration**

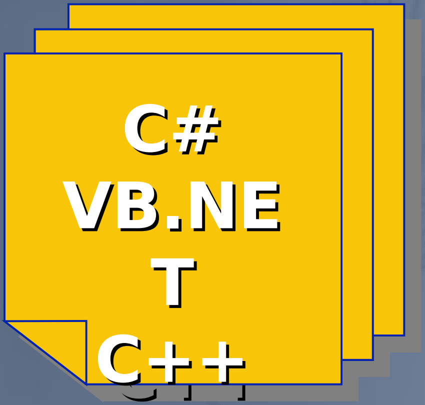
Architectures

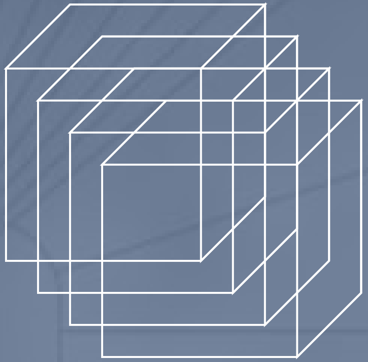
Architecture of J2EE



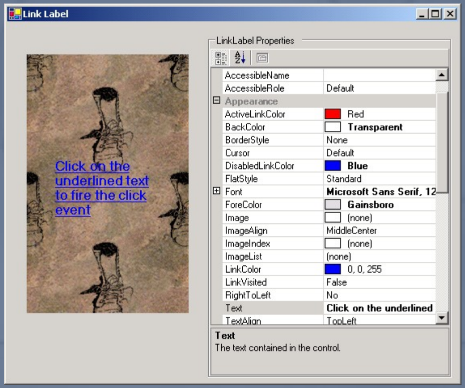
Architectures

Purpose and Structure of .NET





.NET Framework



Link Label

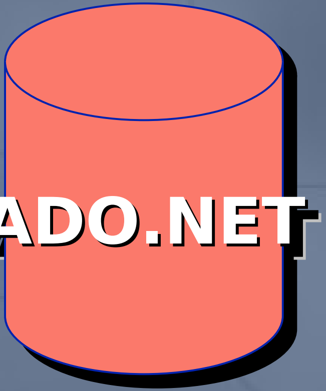
LinkLabel Properties

Click on the underlined text to fire the click event

AccessibleName	
AccessibleRole	Default
⊟ Appearance	
ActiveLinkColor	🟥 Red
BackColor	⬜ **Transparent**
BorderStyle	None
Cursor	Default
DisabledLinkColor	🟦 **Blue**
FlatStyle	Standard
⊞ Font	**Microsoft Sans Serif, 12**
ForeColor	⬜ **Gainsboro**
Image	⬜ (none)
ImageAlign	MiddleCenter
ImageIndex	⬜ (none)
ImageList	(none)
LinkColor	🟦 0, 0, 255
LinkVisited	False
RightToLeft	No
Text	**Click on the underlined**
TextAlign	TopLeft

Text
The text contained in the control.

Forms

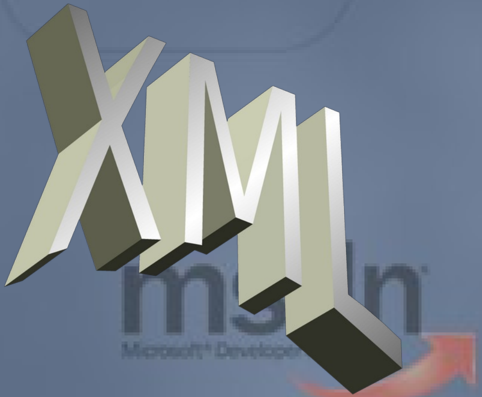


ADO.NET

Common Language Runtime

ASP.NET

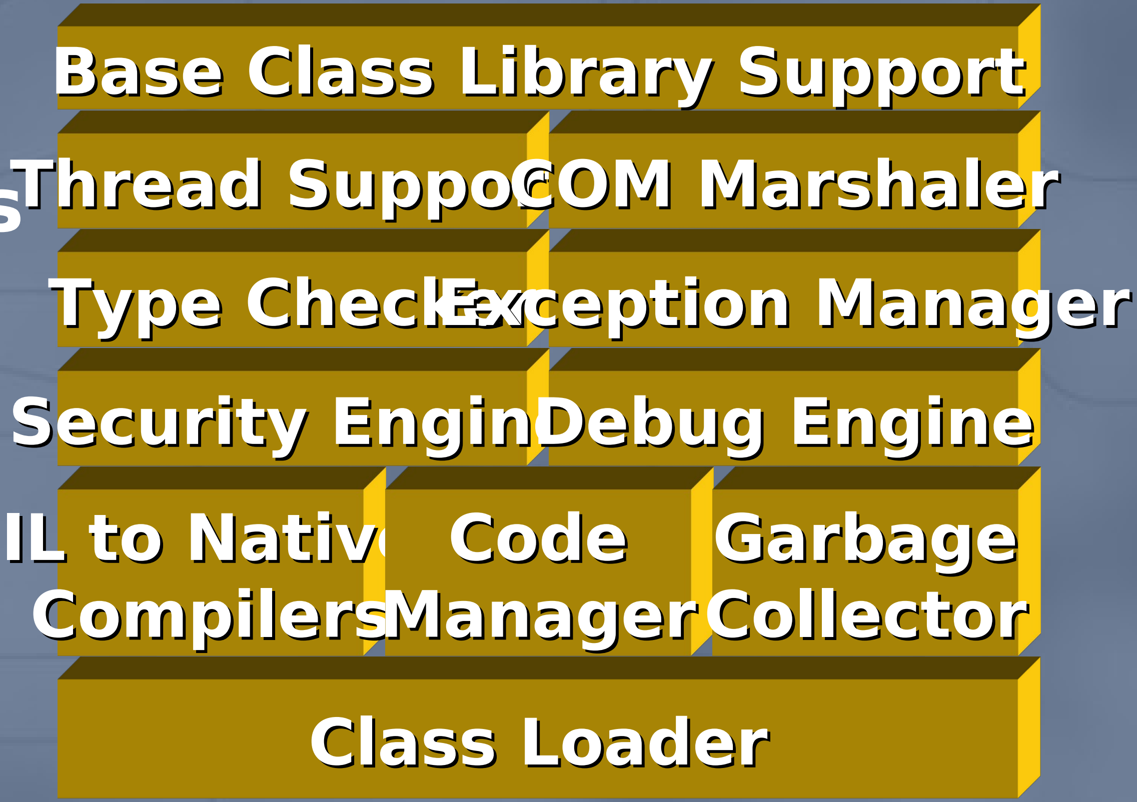
Web Service



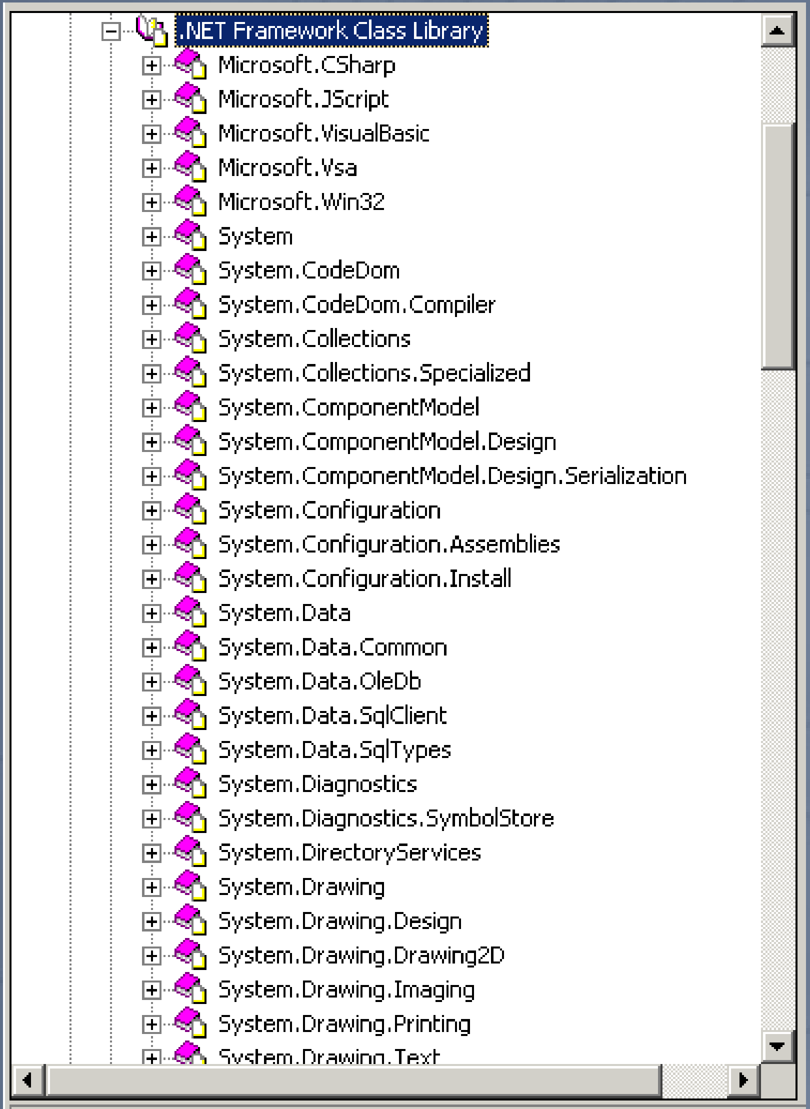
Architectures

Common Language Runtime

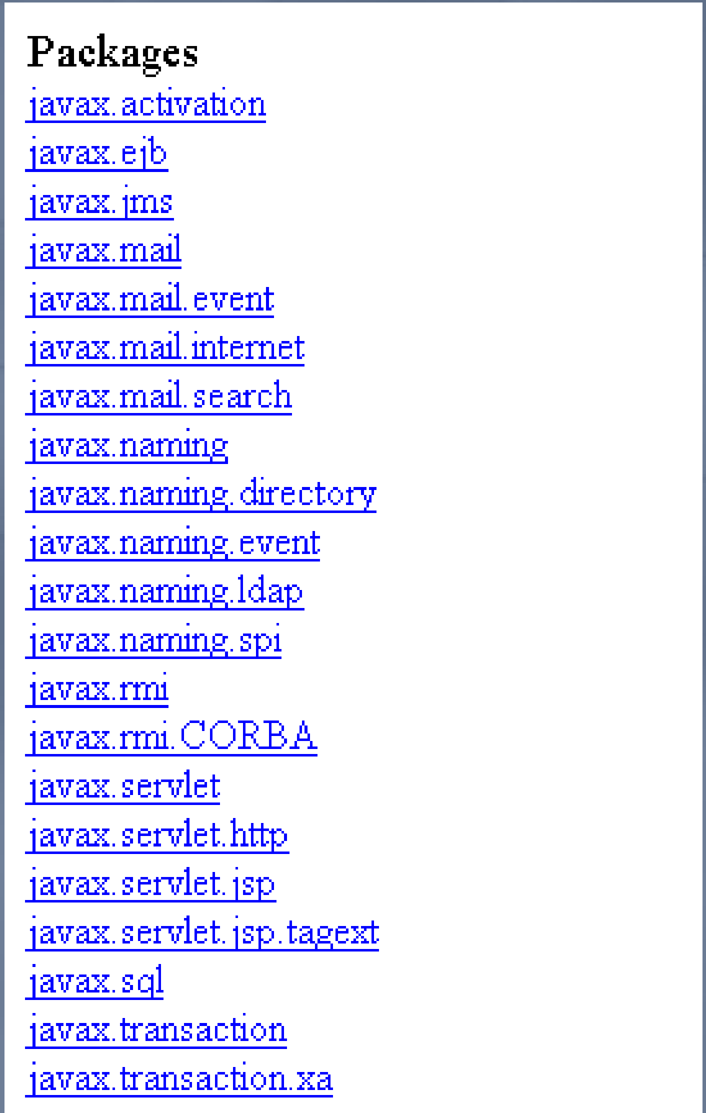
- Assemblies
  - Logical DLLs
- Component Oriented
  - Events
  - Delegates
  - Attributes
  - Properties



Architectures
Frameworks and Packages



.NET Framework Class Library
- Microsoft.CSharp
- Microsoft.JScript
- Microsoft.VisualBasic
- Microsoft.Vsa
- Microsoft.Win32
- System
- System.CodeDom
- System.CodeDom.Compiler
- System.Collections
- System.Collections.Specialized
- System.ComponentModel
- System.ComponentModel.Design
- System.ComponentModel.Design.Serialization
- System.Configuration
- System.Configuration.Assemblies
- System.Configuration.Install
- System.Data
- System.Data.Common
- System.Data.OleDb
- System.Data.SqlClient
- System.Data.SqlTypes
- System.Diagnostics
- System.Diagnostics.SymbolStore
- System.DirectoryServices
- System.Drawing
- System.Drawing.Design
- System.Drawing.Drawing2D
- System.Drawing.Imaging
- System.Drawing.Printing
- System.Drawing.Text



Packages

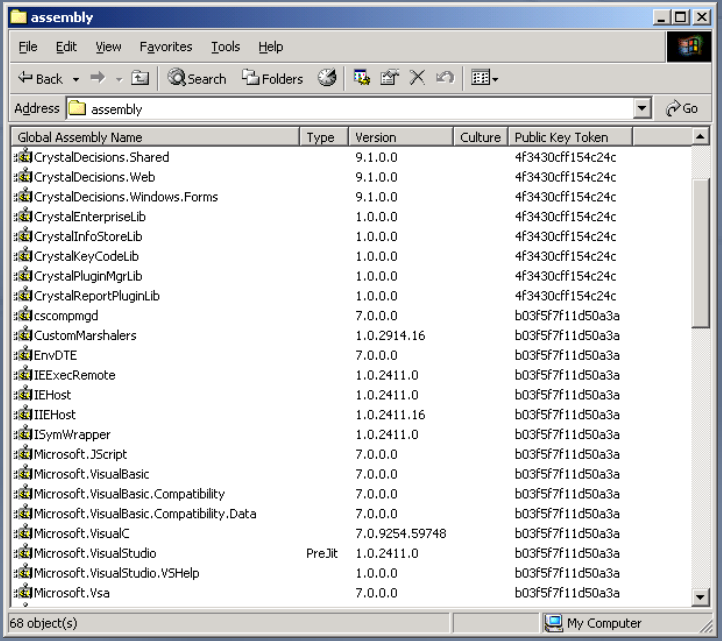
javax.activation
javax.ejb
javax.jms
javax.mail
javax.mail.event
javax.mail.internet
javax.mail.search
javax.naming
javax.naming.directory
javax.naming.event
javax.naming.ldap
javax.naming.spi
javax.rmi
javax.rmi.CORBA
javax.servlet
javax.servlet.http
javax.servlet.jsp
javax.servlet.jsp.tagext
javax.sql
javax.transaction
javax.transaction.xa

Architectures
Directory Services

- Naming and Directory Services
  - Transparent access to distributed objects and services
  - Different vendors implement different native APIs
- Java Naming and Directory Interface
  - Integral part of J2EE, fundamental
  - Open specification
  - Providers for LDAP, DNS, NIS, NDS, RMI, CORBA, …
- .NET System.DirectoryServices Namespace
  - Provides access to Active Directory
  - Providers for IIS, LDAP, NDS, WinNT
  - Less integral to .NET

Architectures
Applications Deployment

- J2EE/EJB
  - Deploy using Application Server
  - Create client packages
  - Updating EJBs can cause incompatibilities
- .NET
  - XCOPY deployment
  - Global Assemblies and the GAC
  - Strong Names avoid conflict
  - Versioning easier



Global Assembly Name	Type	Version	Culture	Public Key Token
CrystalDecisions.Shared		9.1.0.0		4f3430cff154c24c
CrystalDecisions.Web		9.1.0.0		4f3430cff154c24c
CrystalDecisions.Windows.Forms		9.1.0.0		4f3430cff154c24c
CrystalEnterpriseLib		1.0.0.0		4f3430cff154c24c
CrystalInfoStoreLib		1.0.0.0		4f3430cff154c24c
CrystalKeyCodeLib		1.0.0.0		4f3430cff154c24c
CrystalPluginMgrLib		1.0.0.0		4f3430cff154c24c
CrystalReportPluginLib		1.0.0.0		4f3430cff154c24c
cscompmgd		7.0.0.0		b03f5f7f11d50a3a
CustomMarshalers		1.0.2914.16		b03f5f7f11d50a3a
EnvDTE		7.0.0.0		b03f5f7f11d50a3a
IEExecRemote		1.0.2411.0		b03f5f7f11d50a3a
IEHost		1.0.2411.0		b03f5f7f11d50a3a
IIEHost		1.0.2411.16		b03f5f7f11d50a3a
ISymWrapper		1.0.2411.0		b03f5f7f11d50a3a
Microsoft.JScript		7.0.0.0		b03f5f7f11d50a3a
Microsoft.VisualBasic		7.0.0.0		b03f5f7f11d50a3a
Microsoft.VisualBasic.Compatibility		7.0.0.0		b03f5f7f11d50a3a
Microsoft.VisualBasic.Compatibility.Data		7.0.0.0		b03f5f7f11d50a3a
Microsoft.VisualC		7.0.9254.59748		b03f5f7f11d50a3a
Microsoft.VisualStudio	PreJit	1.0.2411.0		b03f5f7f11d50a3a
Microsoft.VisualStudio.VSHelp		1.0.0.0		b03f5f7f11d50a3a
Microsoft.Vsa		7.0.0.0		b03f5f7f11d50a3a

68 object(s) My Computer

Architectures
.NET Security

- Who Can Execute? - Role Based
  - Principals and Identities
  - Programmatic and declarative
- Who Wrote It? - Signing
  - Strong Names and Certificates to guarantee code authenticity
- What Can It Do? - Code Access Security
  - Makes sure a downloaded assembly doesn’t format your C: drive!
  - Administrator can set policy

Architectures
XML in J2EE and .NET

- J2EE
  - Separate package - JAXP
    - API for DOM Level 2, SAX 2.0, XSLT 1.0
    - Requires implementations
- .NET - Built-in, Fundamental
  - XML 1.0 + DTD Support
  - Schemas
  - XPath
  - XSLT
  - DOM Level 2 Core
  - SOAP 1.1

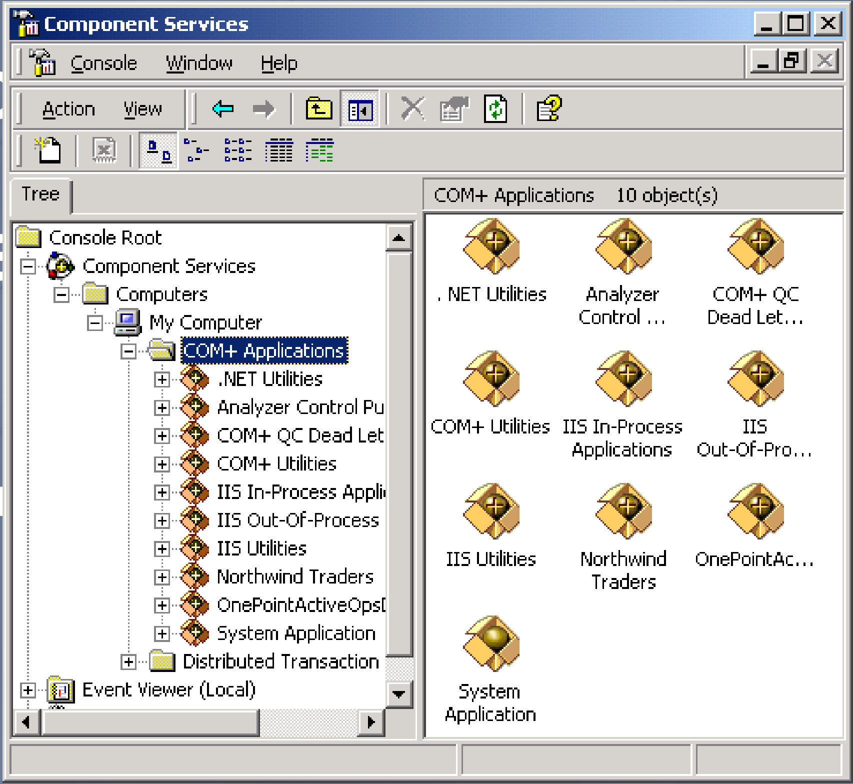
Agenda

- Architectures
- Components And Services
- Distributed Applications
- Accessing Data
- Web Services
- Web Applications
- Application Integration

Components And Services

Serviced Components

- The Easy Way to Build CO Applications
- Support is Integral to .NE
  - System.EnterpriseServices
  - ServicedComponent Class
  - Attributes define behaviour
- Similar to Session EJBs
  - But can also run in-process
  - [ApplicationActivation]
- Dynamic and Manual Registration



Components And Services

Transactions and Object State

- Automatic Transaction Processing - [Transaction]
  - Uses DTC as Transaction Manager
  - SetComplete, SetAbort, EnableCommit, DisableCommit indicate object state
  - [AutoComplete]
  - COM+ Provides context
- Session EJBs
  - Transaction attributes specified at deploy time
  - Container provides context
  - JTS provides Transaction Manager
- Compensating Resource Manager Support
  - Long-running "transactions"
  - Include non-transactional data in a transaction
- State can be Private or Shared
  - Shared Property Groups

Components And Services

Entities and Persistent State

- EJB Model Provides Entity Beans
  - Bridge the Relational/Object gap
  - Manage Persistent State
- Container Managed Persistence
  - EJB Server manages database interactions
  - Abstract database details away from business/bean logic
- Implementations
  - Many are highly optimized for single tables
  - Vendor extensions for parent/child relationships

Components And Services
Object Lifecycle

- **JIT Activation - `[JustInTimeActivation(true)]`**
  - **Objects created on demand**
  - **Destroyed when "done" bit set (transaction complete)**
  - **Client retains reference to dummy stub**
- **Object Pooling - `[ObjectPooling]`**
  - **More control than Session EJBs**
  - **Minimum/Maximum Pool Size properties**
  - `CanBePooled`, `Activate`, `Deactivate` **methods**
  - **Use if activation quicker than creation**
- **J2EE Session EJBs**
  - **Similar model**
  - **Less comprehensive**

Components And Services

Asynchronous Activation and Events

- Queued Components
  - Asynchronous activation
  - Based on MSMQ
  - `[ApplicationQueuing(Enabled=true, QueueListenerEnabled=true)]`
  - `[InterfaceQueueing]`to tag queued interfaces
- Loosely-Coupled Events
  - Publisher/Subscriber metaphor
  - More efficient than repeatedly polling a server
  - Publisher and Subscriber must both extend ServicedComponent
- J2EE
  - Message Driven EJBs (1.3)
  - Use JMS (1.2)

Components And Services

Roles and Security

- Integrated With Windows® Security
  - Parallel to .NET role-based security (use one or the other)
- Declarative Security with Attributes
  - `[ApplicationAccessControl]` permits security configuration
  - `[ComponentAccessControl]` enables security checking
  - `[SecurityRole]` to define roles and associate users/groups
- Programmatic Security
  - `SecurityCallContext` Class
  - `IsSecurityEnabled` property
  - `IsCallerInRole` method
- J2EE
  - EJBs have equivalent functionality
  - Declarative security configured at deploy time
  - May not be as closely integrated with the Operating System

Components And Services

Component Services versus EJB

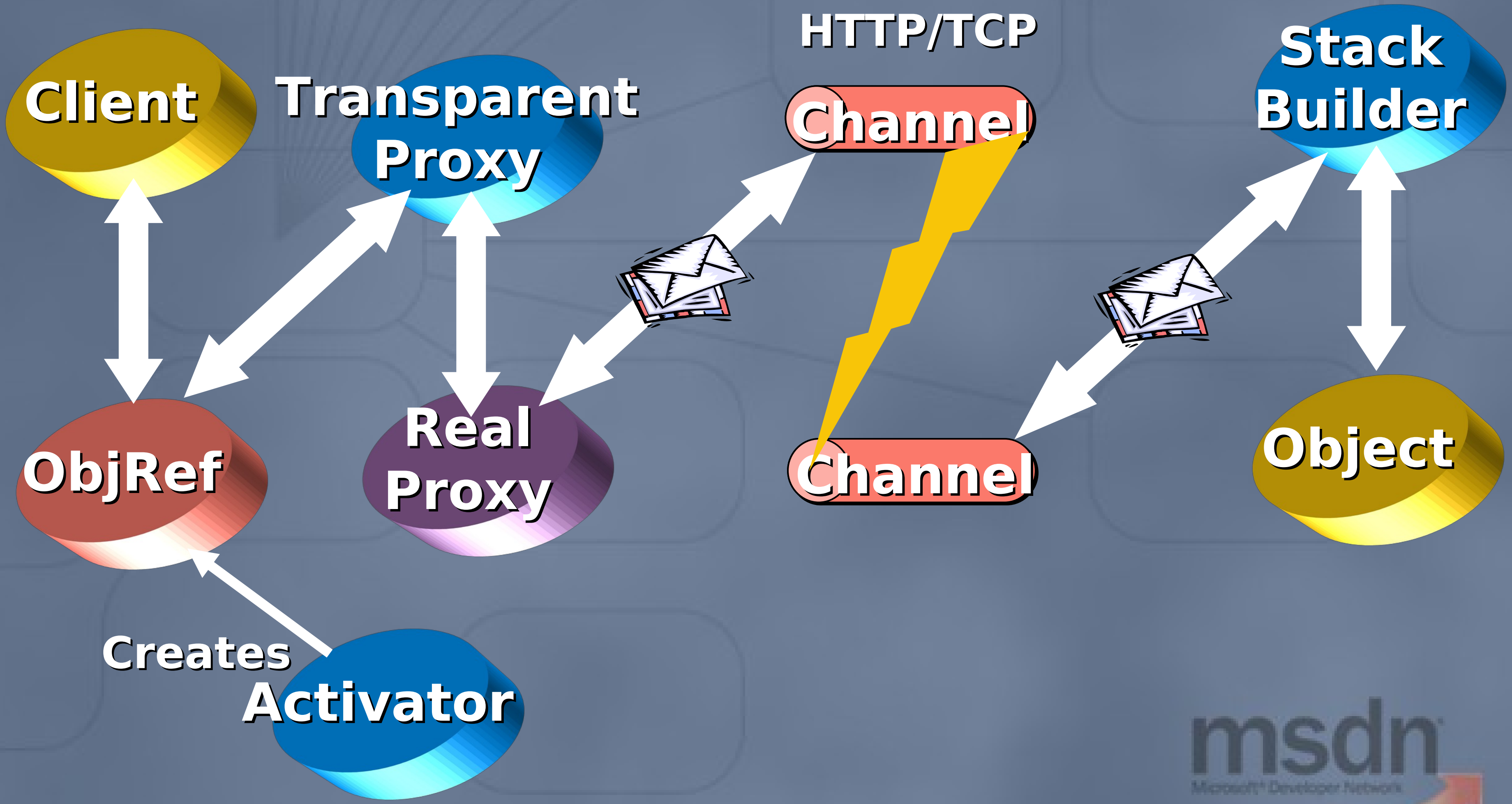
- EJB
  - EJB Server creates EJB Container
  - EJB Container manages EJBs
  - Provides context, security, transactions, pooling, managed persistence
  - Layered above Operating System
- Component Services
  - COM+ Base Services supply application host (server)
  - Provides context, security, transactions, pooling, loosely-coupled events, asynchronous activation
  - Integrated into Operating System – highly tuned

Agenda

- Architectures
- Components And Services
- Distributed Applications
- Accessing Data
- Web Services
- Web Applications
- Application Integration

Distributed Applications

.NET Remoting



Distributed Applications

Processing and Marshaling

- Marshal By Value
  - `[Serializable]` attribute
  - `ISerializable` interface
  - Value copy in client
- Marshal By Reference
  - Inherit from `MarshalByRefObject`
  - Remote object reference in client
- Message Sinks
  - `IMessageSink` object chains on client and server
  - Custom processing and transformations
- Remote Events Supported

Distributed Applications

Data Transmission

- Messages are Serialized for Transportation
  - Channel sends/receives data
- Formatter Sinks
  - Binary for TCP
  - SOAP/XML for HTTP - interoperable
  - Create your own - `IRemoteFormatter` interface
- Stack Builder Sink
  - Creates a Stack frame
  - Invokes the target object method
  - Passes return values back through the Channel to the client
  - Catches exceptions and returns them to the client

Distributed Applications

Activation Modes

- Server Activation
  - Host application creates end-point
  - Singleton and SingleCall modes
- Client Activation
  - Remote object created on demand by client
  - Client controls object lifetime
  - Uses Lifetime Leases to aid garbage collection
  - Leasing Distributed Garbage Collector (LDGC)
- HTTP Objects Accessed Through IIS

Distributed Applications

Security

- .NET Code Access Security Controls Marshaled Objects
- User Authentication and Authorization
  - Implement a custom Message Sink
  - Prefer HTTP channels to TCP
  - Can use integrated security with IIS
- RMI
  - Built-in Java Security
  - Security Manager/Sandbox
  - Standard or Customized

Distributed Applications

.NET Remoting versus RMI

- .NET Remoting
  - Based upon common standards and protocols
  - Works as-is, but highly extensible and securable
  - Custom routing and marshaling can aid scalability
  - Supports synchronous and asynchronous method invocation
  - Can interoperate with other systems
- RMI/IIOP
  - Targeted at Java
  - Multi-language interoperability through CORBA
  - Requires bootstrap naming service (RMI Registry) or JNDI
  - Extensible using Socket Factories and Custom Sockets

Distributed Applications
Message Queues

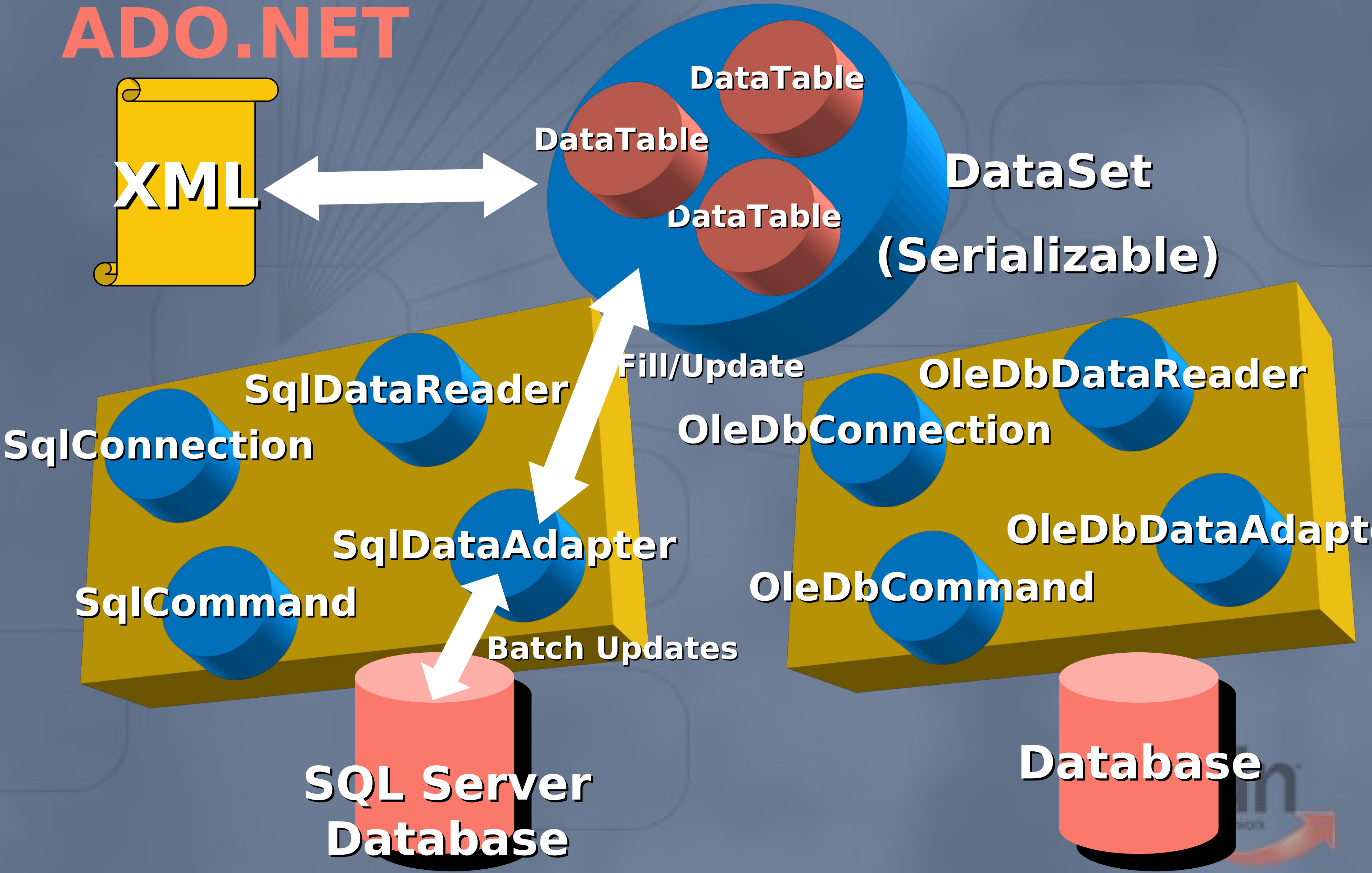
- .NET
  - Asynchronous, persistent messaging using MSMQ
  - Provides events, priorities, transactional semantics, security, custom message formats (XML by default)
- J2EE
  - Java Messaging Service (JMS) API
  - Comprehensive, but complex
  - Open, but requires a JMS provider

Agenda

- Architectures
- Components And Services
- Distributed Applications
- Accessing Data
- Web Services
- Web Applications
- Application Integration

Accessing Data

ADO.NET



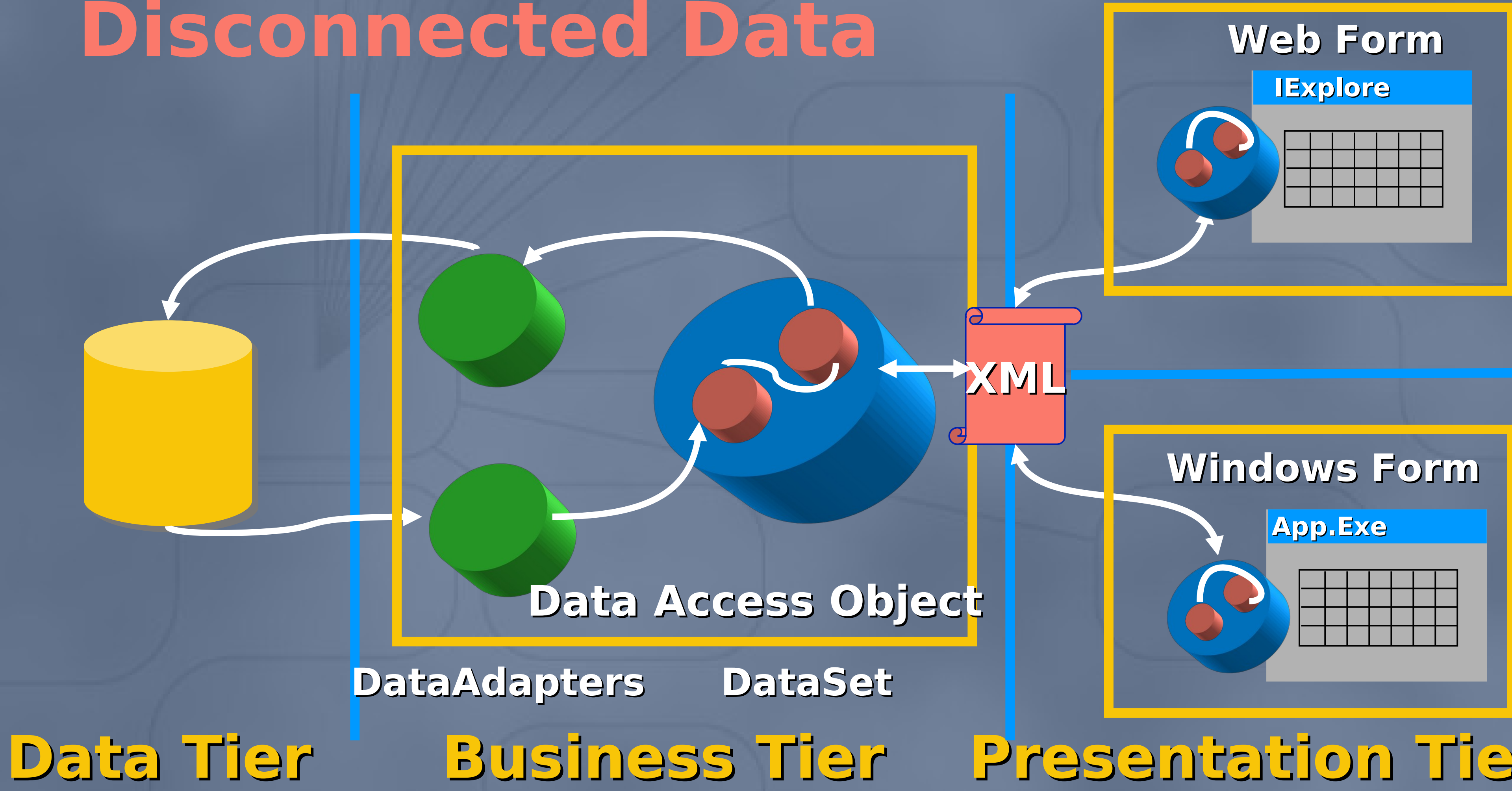
Accessing Data

Firehosing

- DataReader Component
  - Lightweight, fast, forward-only , read-only, row-at-a-time stream
  - Blasts data from source to application
  - Minimal locking (if any), improved concurrency
  - Extremely efficient means of sequential data access for local applications
- Use separate Command components to update data

Accessing Data

Disconnected Data



Accessing Data

Batch Updates

- Edit/Insert/Delete DataRow(s) In DataTable(s) In DataSet
- Generate DataSet Containing Changes Only
  - `DataSet.GetChanges`
- Validate Changes
  - `DataTable.GetErrors`
- Correct Errors
  - `DataTable.GetErrors, DataRow.GetColumnsInError, DataRow.GetColumnError`
- Create DataAdapter
  - `UpdateCommand, InsertCommand, DeleteCommand`
- Call `Adapter.Update`
  - Optimistic concurrency by default
  - Events to check/trap errors

Accessing Data

Data Binding

- DataBound Controls In .NET Framework
  - Windows Forms and Web Forms controls
  - DataGrid, DataList, Repeater, …
- Can Attach ADO.NET Data Source
  - Programmatically
  - At design-time, with Wizards that generate Connection, Command, Adapter, DataSet
- Can Also Bind To Arrays, Collections, XML

Accessing Data
ADO.NET versus JDBC2

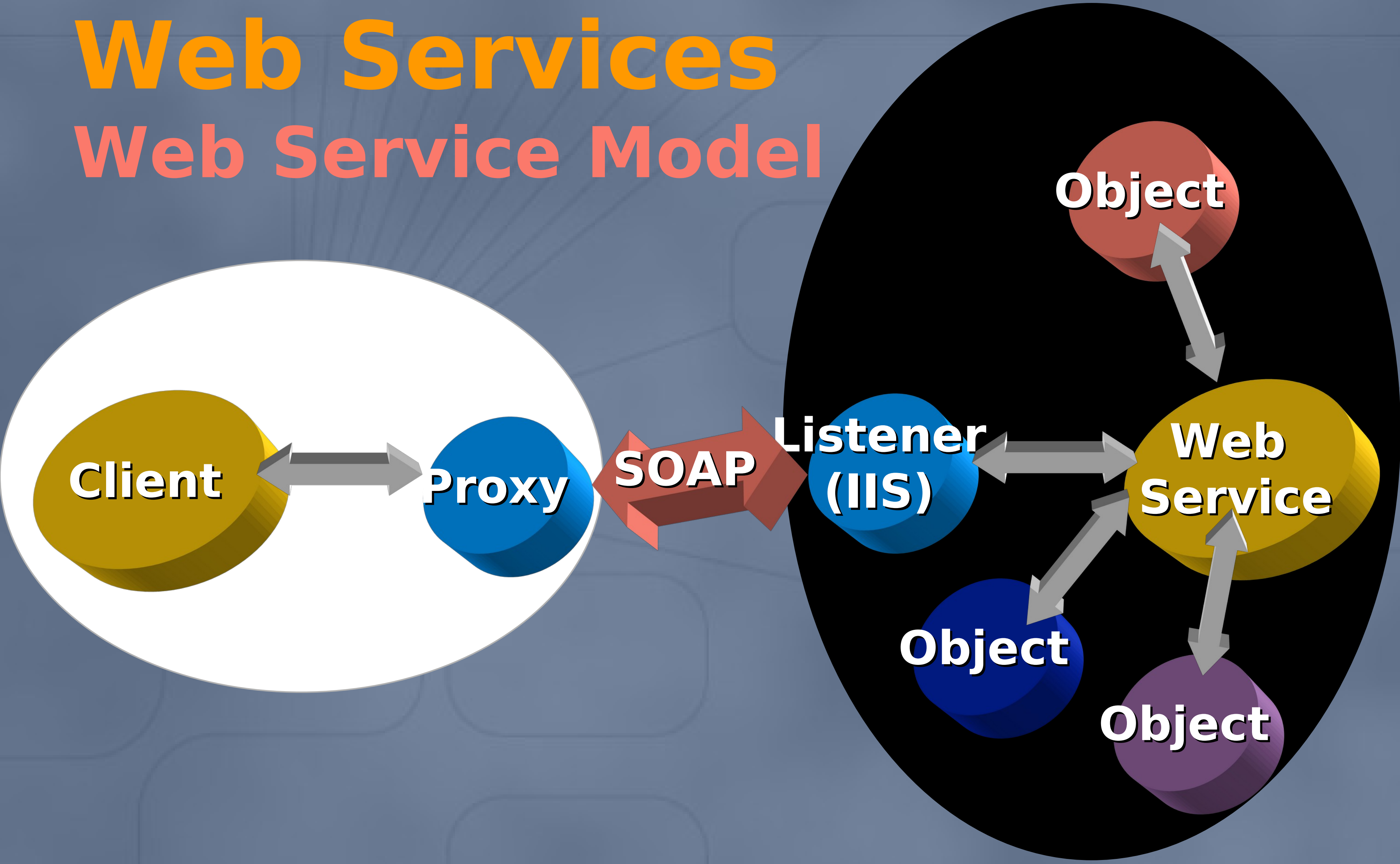
- JDBC Model Based on ODBC
  - Familiar, but requires more work to use
  - SQL oriented (what about non-SQL sources?)
  - Different levels of compliance, performance
- JDBC2 RowSets
  - Bigger and heavier than DataSets
  - More data to marshal
  - More complex to code
- Implementations of Pooled Connections Use JNDI
- XML Data Binding (JAXB)
  - Object serialization/deserialization only, not presentation
  - “Coming soon”
  - Not currently a core part of J2EE

Agenda

- Architectures
- Components And Services
- Distributed Applications
- Accessing Data
- Web Services
- Web Applications
- Application Integration

Web Services

Web Service Model



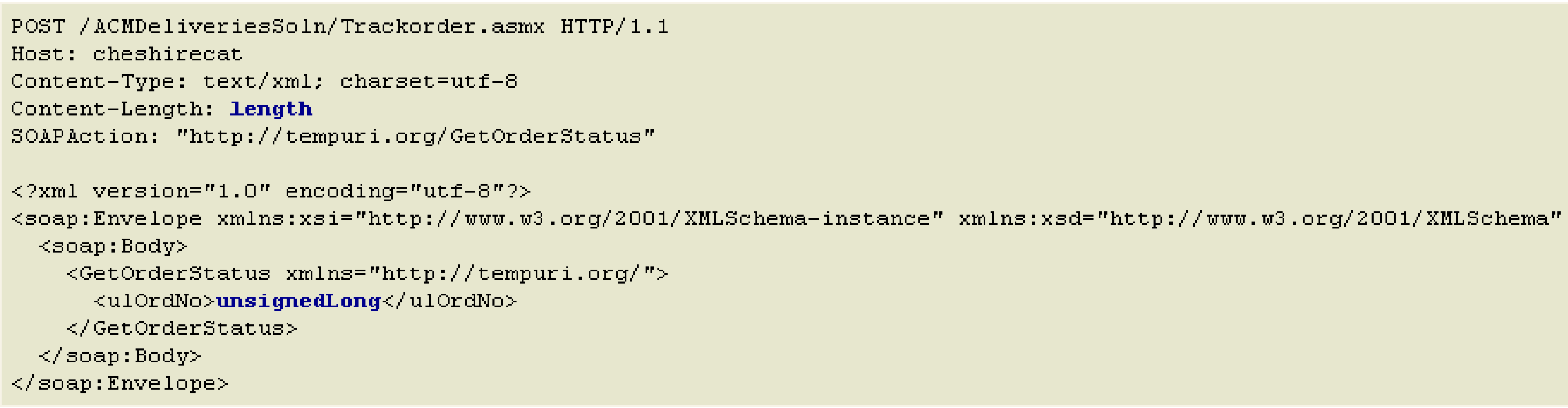
Web Services
.asmx Files

```
<% @ WebService Class=“TrackOrder” %>
Using System;
Using System.WebServices;

public class TrackOrder : WebService
{
  [WebMethod]
  public string GetOrderStatus(ulong ulOrdNo)
  {
    . . .
    return Status;
  }
}
```

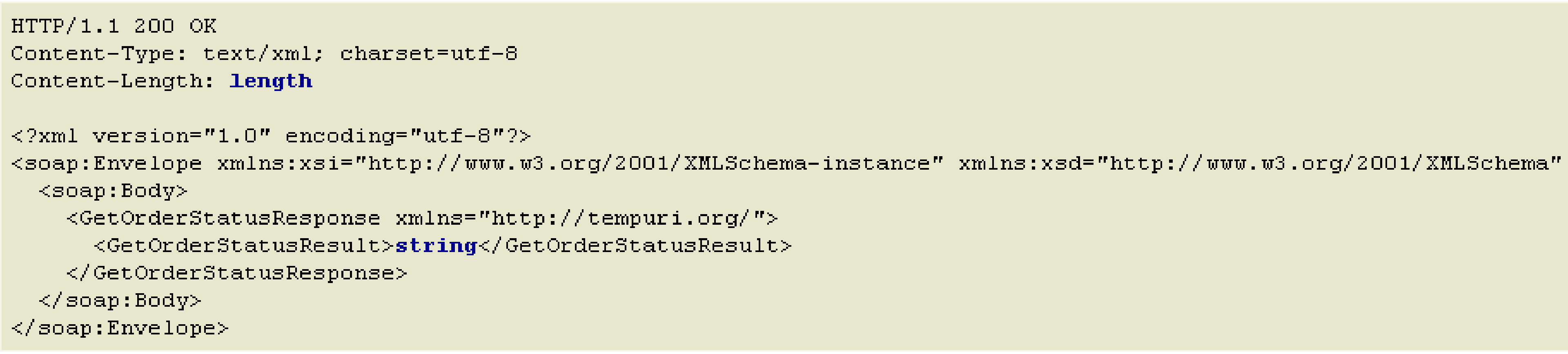
Web Services

SOAP



```
POST /ACMDeliveriesSoln/Trackorder.asmx HTTP/1.1
Host: cheshirecat
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/GetOrderStatus"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  <soap:Body>
    <GetOrderStatus xmlns="http://tempuri.org/">
      <ulOrdNo>unsignedLong</ulOrdNo>
    </GetOrderStatus>
  </soap:Body>
</soap:Envelope>
```



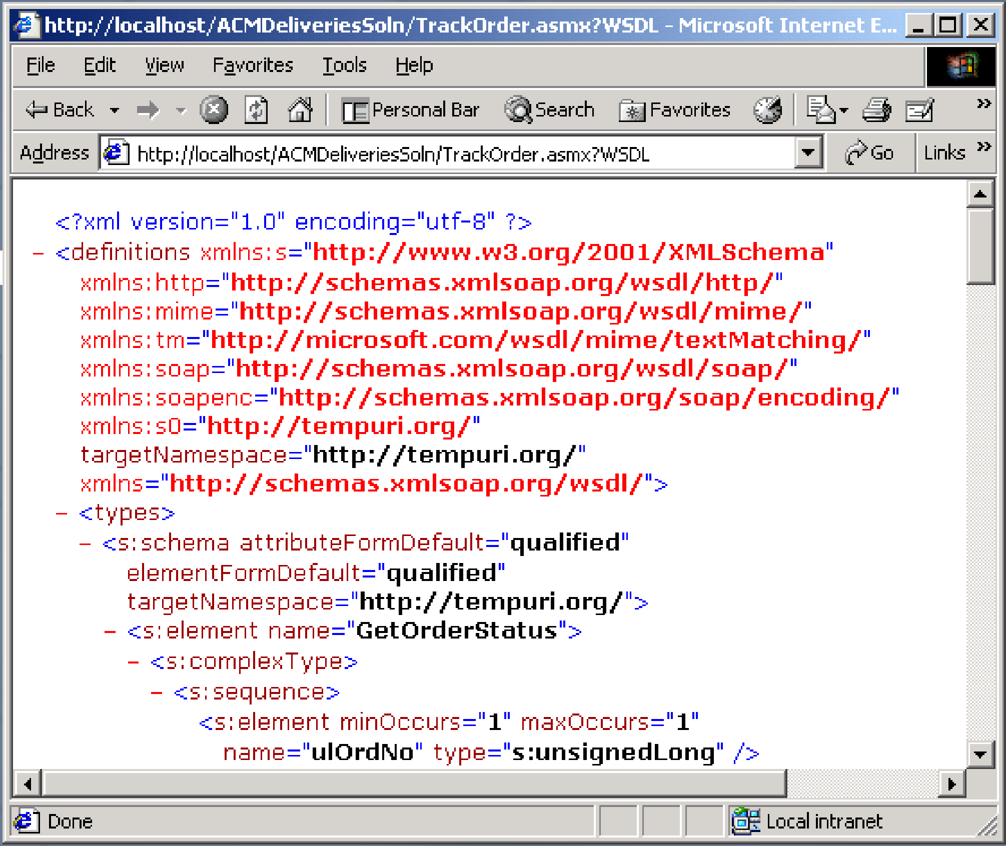
```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  <soap:Body>
    <GetOrderStatusResponse xmlns="http://tempuri.org/">
      <GetOrderStatusResult>string</GetOrderStatusResult>
    </GetOrderStatusResponse>
  </soap:Body>
</soap:Envelope>
```

Web Services

Web Services Description Language

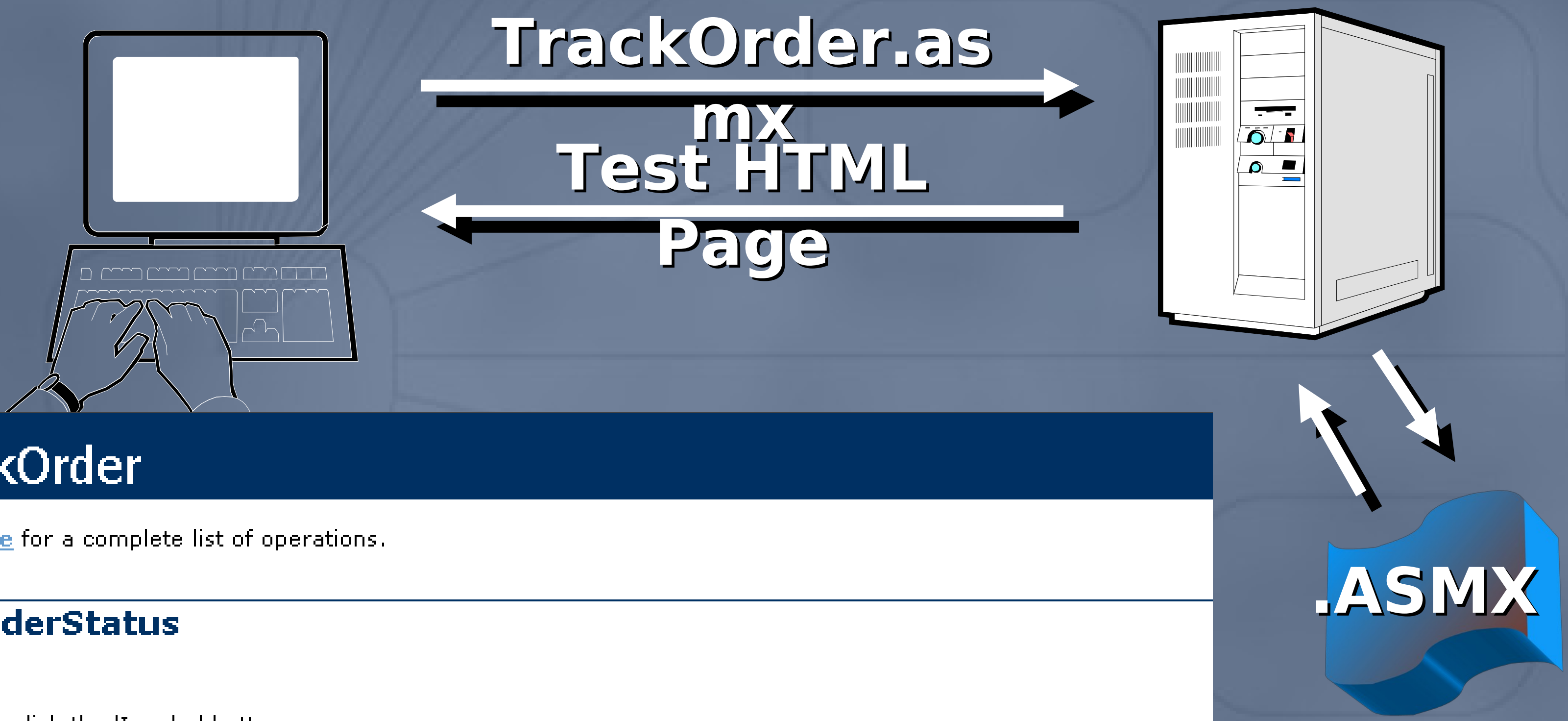
- A Web Service Can Be Asked For A List Of Its Methods
  - It should respond with a description in an understood format
- WSDL Is A Standard Format For Describing Networked XML Services
  - Useful for automating communications between Web Services
  - Orchestration

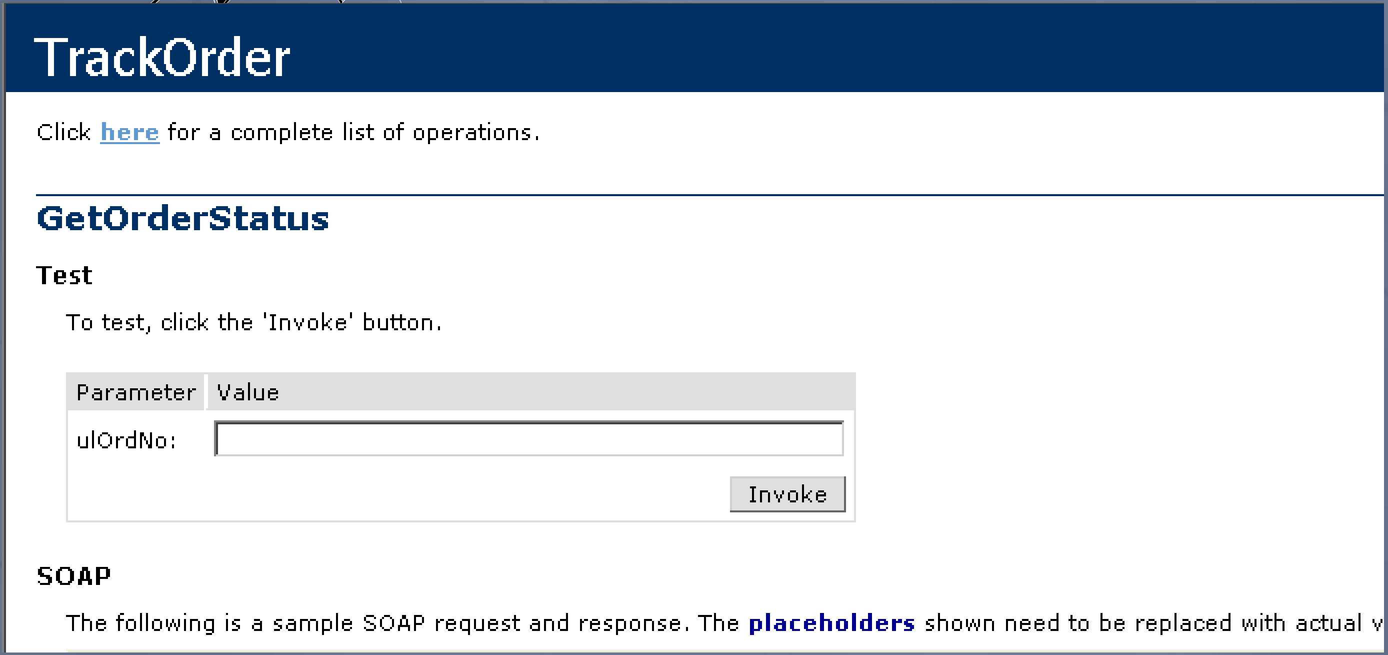

http://localhost/ACMDeliveriesSoln/TrackOrder.asmx?WSDL - Microsoft Internet E...
File Edit View Favorites Tools Help
Back Personal Bar Search Favorites
Address http://localhost/ACMDeliveriesSoln/TrackOrder.asmx?WSDL
Go Links
Done
Local intranet

```
  <?xml version="1.0" encoding="utf-8" ?>
- <definitions xmlns:s="http://www.w3.org/2001/XMLSchema"
    xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
    xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
    xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:s0="http://tempuri.org/"
    targetNamespace="http://tempuri.org/"
    xmlns="http://schemas.xmlsoap.org/wsdl/">
  - <types>
    - <s:schema attributeFormDefault="qualified"
        elementFormDefault="qualified"
        targetNamespace="http://tempuri.org/">
      - <s:element name="GetOrderStatus">
        - <s:complexType>
          - <s:sequence>
              <s:element minOccurs="1" maxOccurs="1"
                name="ulOrdNo" type="s:unsignedLong" />
```

Web Services

Testing





TrackOrder

Click here for a complete list of operations.

GetOrderStatus

Test

To test, click the 'Invoke' button.

Parameter	Value
ulOrdNo:	

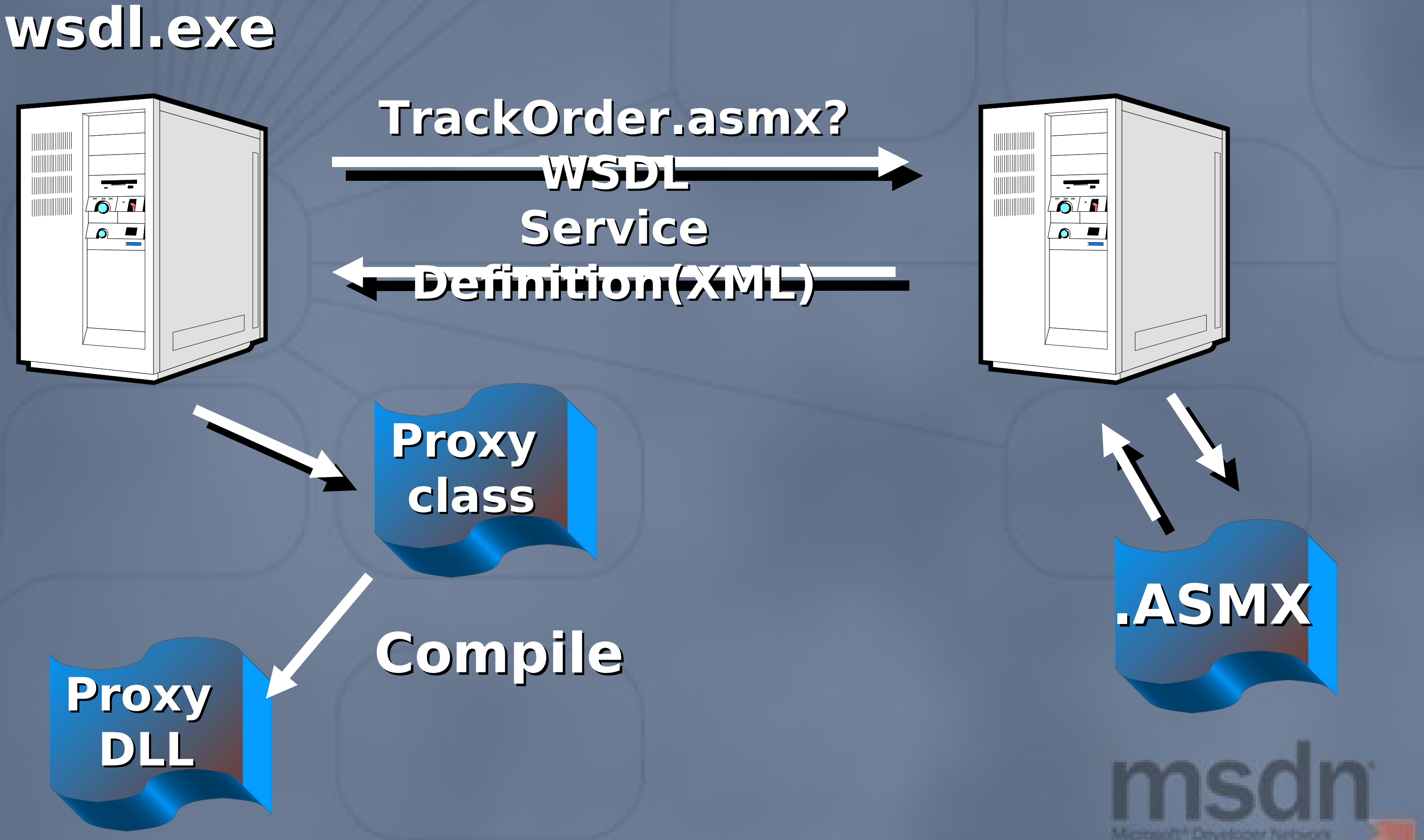
Invoke

SOAP

The following is a sample SOAP request and response. The **placeholders** shown need to be replaced with actual v

Web Services

Web Service Proxies and Clients



Agenda

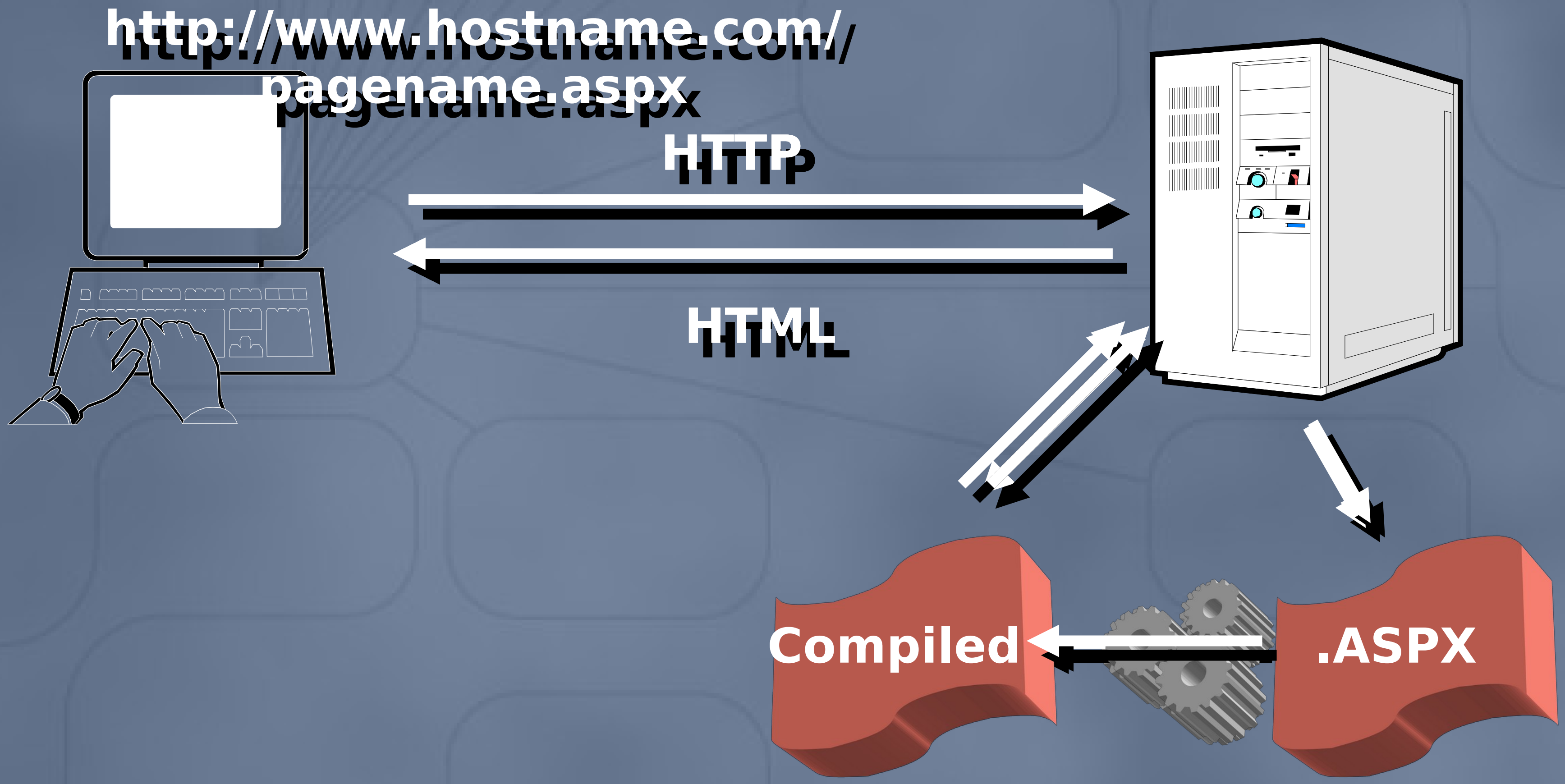
- Architectures
- Components And Services
- Distributed Applications
- Accessing Data
- Web Services
- Web Applications
- Application Integration

Web Applications
ASP.NET

- Updated ASP, For .NET
- Separates Presentation/Business Logic
  - Code Behind Forms
  - C#, VB.NET, JScript
- Web Forms
  - Quicker development
  - Powerful controls
  - Event driven programming
  - State preservation

Web Applications

ASP.NET Architecture



Web Applications
HTML and Server Controls

- Detects Client Browser Capabilities
  - Generates appropriate HTML
- Runat
  - enables server-side processing

HTML

```
<div id="MyDiv" runat="server"/>
```

Server Controls

```
<asp:TextBox id="txtUserName"
             runat="server"/>
<asp:Button OnClick="SubmitBtn_Click"
            runat="server"/>
```

Web Applications

Data Binding

- Server Controls Bind To Many Data Sources
  - Collections

```
<asp:ListBox id="List1" datasource='<%# myArray %>'
runat="server">
```

  - ADO.NET
    - DataReader
    - DataSet
  - XML
- DataBinding Expressions In ASP.NET

```
<%# GetBalance(custID) %>
```

msdn
Microsoft® Developer Network

Web Applications

Events

```
<script language="C#" runat="server">
  void SubmitBtn_Click(Object sender, EventArgs e)
  {
   Response.Write ("Hello " + txtUserName.Text);
  }
</script>
<body>
 <form runat="server">
  <asp:TextBox ID="txtUserName" runat="server"/>
  <asp:Button OnClick="SubmitBtn_Click" Text="Submit"
              runat="server"/>
 </form>
</body>
```

Web Applications

Data Controls

- DataGrid, Repeater and DataList
  - HTML templates
  - DataGrid and DataList updateable

```
<asp:DataList id="dataList1"
  runat="server"
  RepeatColumns="3"
  GridLines="Both"
  CellPadding="4"
  CellSpacing="0" >
  <ItemTemplate>
    Order Date:
      <%# DataBinder.Eval(Container.DataItem, "DateTimeValue") %>
    Quantity:
      <%# DataBinder.Eval(Container.DataItem, "IntegerValue") %>
  </ItemTemplate>
</asp:Datalist>
```

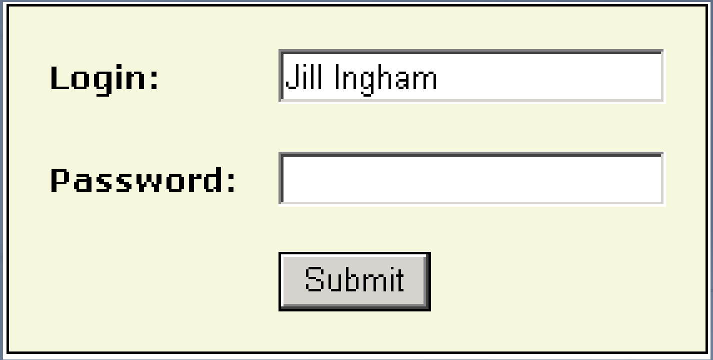
Web Applications
Validation Controls

```
<asp:RangeValidator
id="rangeValString"
Type="String"
ControlToValidate="txtAnimal"
MaximumValue="Zebra"
MinimumValue="Aardvark"
runat="server"/>
```

- Validate User Input
  - RequiredFieldValidator
  - RangeValidator
  - CompareValidator
  - RegularExpressionValidator
  - CustomValidator
  - ValidationSummary
- Generate Code
  - Client-side JavaScript if browser capable
  - Server-side checking always performed

Web Applications
User Controls

- Custom Web Controls
  - Just like VB6 controls!
- Generate HTML output
- Can Inherit



Login:	Jill Ingham
Password:	

```
<%@ Register TagPrefix="Acme" TagName="Login" Src="login.ascx" %>
<html>
  ...
  <form runat="server">
    <Acme:Login id="MyLogin"
      UserId="Jill Ingham"
      Password="Secret"
      BackColor="beige"
      runat="server"/>
  </form>
</html>
```

Web Applications
Caching

- Application Cache
  - Stores objects in memory
  - Cleared when application terminates
- Lifetime Control
  - Scavenging
  - Expiration
  - Dependencies

```
Cache.Insert("MyData", Source, ...,
     DateTime.Now.AddHours(1), ...)
```

```
myValue = Cache["mykey"];
if(myValue != null )
{
  DisplayData(myValue);
}
```

Web Applications

State Management

- ASP.NET Web Forms Preserve State Between Posts
  - Can be disabled to save bandwidth
- Application/Session Objects
  - Can use Session/Application events to load/save Session/Application data
    - Threading issues
    - Stored In Server Process
- Web Farms
  - More scalable
  - Store state in StateServer or SQL Server™

```
<sessionState
  cookieless="true"
  mode="StateServer"
  stateConnectionString="tcpip=localhost:42424" />
```

Web Applications
ASP.NET versus JSP/Servlets

- Clearer Separation Of Business/Presentation Logic Than Servlets
- Less Overhead Than JSP
  - No beans, tags
- Easier To Develop
  - Code Behind Forms
  - Server controls generate HTML automatically
  - Data binding reduces code
  - Reusable presentation logic
- Scalability
  - State management can span servers
  - Built into ASP.NET, usually available with most “serious” J2EE servers
- Deployment
  - Just XCOPY

Agenda

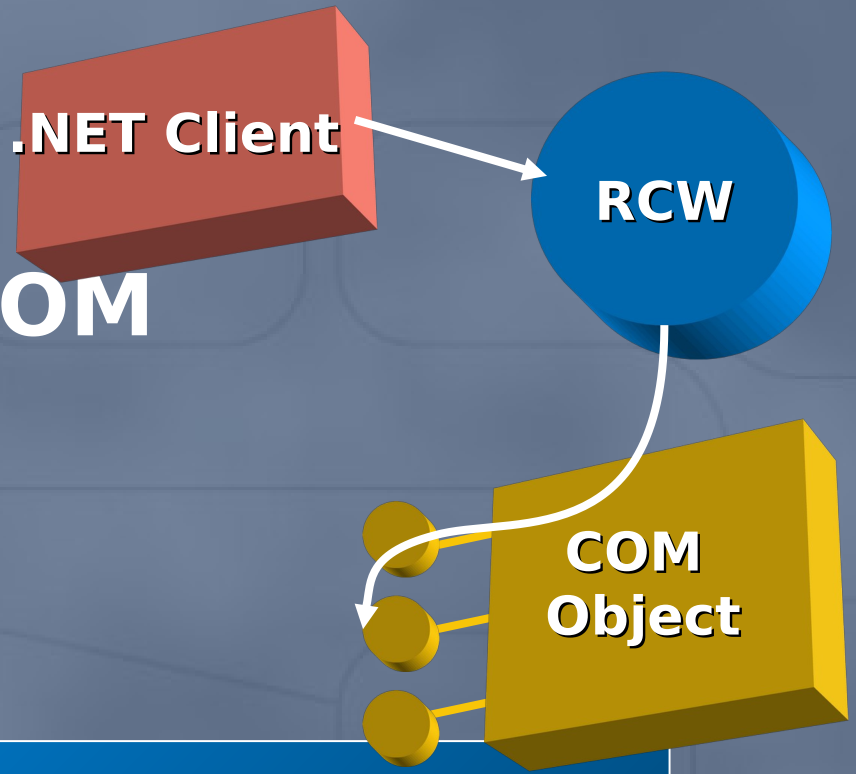
- Architectures
- Components And Services
- Distributed Applications
- Accessing Data
- Web Services
- Web Applications
- Application Integration

Application Integration
Integrating Legacy Code

- Large Base Of Existing Code
  - COM Servers
  - Libraries, DLLs
- Preserve Investment
  - Need to call from .NET
  - Useful to Invoke .NET components from unmanaged environment
- .NET Interop Features
  - Runtime Callable Wrappers (RCW)
  - COM Callable Wrappers (CCW)
  - Platform Invocation Services (PInvoke)
  - Windows Service Applications
- Java Interop Features
  - JNI – less platform specific
  - Invoking Java from non-Java in process is a challenge
  - Java Connector Architecture (J2EE 1.3)

Application Integration

Interop with COM



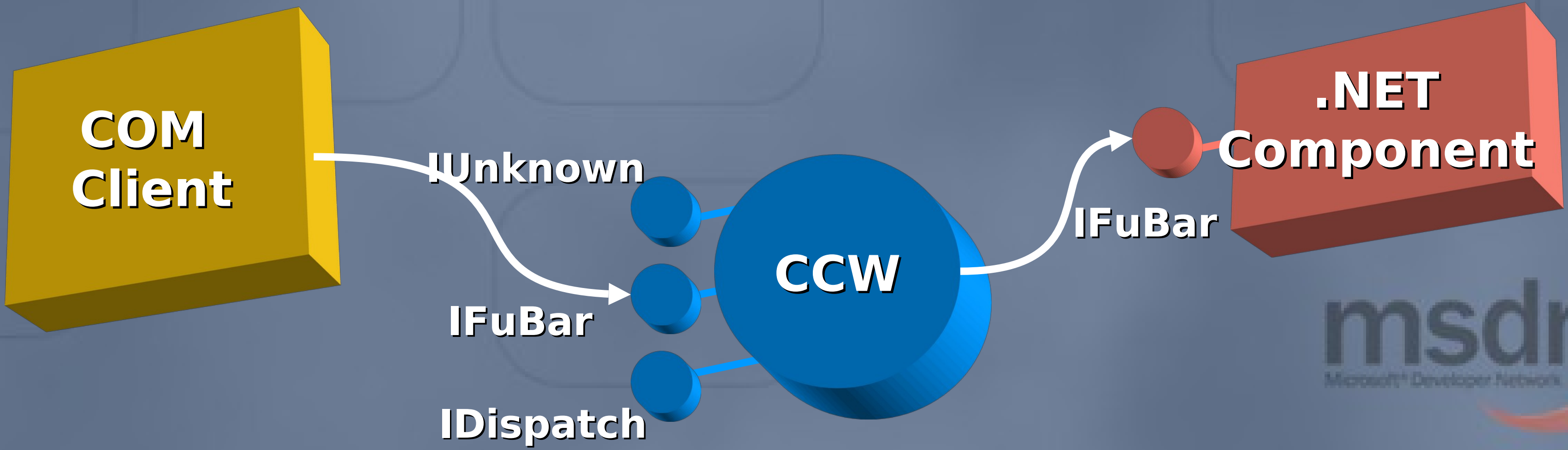
- Early Binding To COM
  - Create RCW (TlbImp.Exe)
  - Use as proxy
- Late Binding

```
System.Type MyType;
Object MyObject;
MyType = Type.GetTypeFromProgID(“MyProg.Id”);
MyObject = Activator.CreateInstance(MyType);
MyType.InvokeMember(“MyMethod”, ..., MyObject, ...);
```

Application Integration

Packaging a .NET Component

- Calling .NET Component From COM
  - Use interfaces, mark with `[ComVisible(true)]`
  - Create strong name (Sn.Exe)
  - Create CCW (TlbExp.Exe)
  - Register component (RegAsm.Exe)



Application Integration
Platform Invocation Services

```
using System.Runtime.InteropServices;
[StructLayout(LayoutKind.Sequential)]
public struct SystemTime {
  public ushort wYear;
  ...
}

public class TestPInvoke {
  [DllImport("Kernel32.dll")]
  public static extern void GetSystemTime(
                                  ref SystemTime sysTime);

  public static void Main() {
    SystemTime sysTime = new SystemTime();
    GetSystemTime(ref sysTime);
    ...
  }
}
```

Application Integration
Windows Service Applications

- Windows Services Written In .NET
  - Daemon processes – can auto-start on system boot
  - Run in their own context, using supplied security credentials
  - Closely coupled to the OS
  - Can define custom Event Log processing
- To Create
  - Use pre-supplied template with Visual Studio.NET
  - Or
    - Inherit from System.ServiceProcess.ServiceBase
    - Override OnStart, OnStop, OnPause, OnContinue, OnShutdown methods
  - Define Installer
  - Install service using InstallUtil.Exe
- ServiceController Components
  - Connect to and control the behaviour of existing Services

Session Summary

- The Highlights Of .NET Compared To J2EE
  - Distributed applications
  - Remoting and marshalling
  - In-process and out-of-process architectures
  - Web clients and servers
  - Synchronous and asynchronous communications
  - Applications deployment and integration
  - Security, scalability and efficiency

For More Information…

- MSDN Web Site at
  - msdn.microsoft.com
- Microsoft .NET Information Sites at
  - www.microsoft.com/net
  - msdn.microsoft.com/vstudio/nextgen/technology
- Additional Information Sites at
  - www.gotdotnet.com
  - www.asp.net
  - www.theserverside.com

Where do you want to go today?®

msdn

Microsoft®